Peris-Lopez, P., Martín, H. (2017). Hardware Trojans against virtual keyboards on e-banking platforms – A proof of concept. AEU - International Journal of Electronics and Communications, vol. 76, pp. 146-151. Available in https://doi.org/10.1016/j.aeue.2017.04.003

Short communication

Hardware Trojans against virtual keyboards on e-banking platforms – A proof of concept

Pedro Peris-Lopez [a,b,*], Honorio Martín [c]

[a] Department of Computer Science and Engineering, University Carlos III of Madrid, Spain
[b] Department of Computer Science, Aalto University, Finland
[c] Department of Electronic Technology, University Carlos III of Madrid, Spain

A B S T R A C T

In the last years there has been a considerable growth on the number of users in on line banking (Szopinski, 2016). Banks must implement strong security solutions and users have to feel safe about the security offered. To securize the users' access, virtual keyboards are commonly used. Unlikely, virtual keyboards are vulnerable to shoulder surfing and malicious software based attacks such as malware and Trojans (Nadkarni et al., 2011; Sapra et al., 2013). In this article we propose a Hardware Trojan (HT), which targets a VGA display and is able to reveal the private information clicked by the user on a virtual keyboard. This HT is very harmful since it defeats the countermeasures (e.g., keyboard mutation or obfus cation) generally used to combat malicious pieces of software (Nayak et al., 2014; Parekh et al., 2011; Rajarajan et al., 2014).

Keywords:
Hardware Trojans
VGA display
On-line banking
Virtual keyboards

1. Introduction

Today on line banking has become one of the most popular methods to perform financial transactions since it offers a win win situation for banks and customers. Indeed on line services have been promoted by banks because of the cost savings and cus tomers' satisfaction [1]. On the other hand, e banking applications offer users the possibility of operating on their accounts from any where at any time.

In spite of the popularity of on line banking services, many cus tomers are reluctant to use these services because of their concerns about the security [7,8]. People who are not particularly technical skilled are often apprehensive about the security level offered by the on line services. Security in the physical world is much more intuitive and tangible for most people (e.g., keep your coordinate cards in a safe place or impede someone can have a gander at the display when you are entering your PIN code [9,10]).

Banks around world make considerable efforts to prevent fraud in their on line platforms. These efforts turn out in authentication solutions that must be economically viable and user friendly [11,12]. Authentication solutions based on user names and pass words are commonly adopted. Passwords typically include a com bination of alphanumeric strings and special characters. This approach often leaves the full responsibility in the hands of cus tomers since they should choose strong passwords.

Even strong passwords, which can not be easily guessed, are under threat due to existing password attacks [13]. The most widely used attacks are keyloggers in their different forms. Keylog gers are known under different names: tracking software, com puter activity monitoring software, keystroke monitoring systems, keystroke recorders, keystroke loggers, keyboard sniffers, and snoopware [14,15]. In order to overcome keyloggers, banks have included virtual keyboards in their e banking applications.

A virtual keyboard is a software component that displays on the screen a visual keyboard with all the standard keys or just a por tion like the numeric keypad. In on screen keyboards, keys are selected using the mouse or another pointer device [6]. Fig. 1 shows two examples of virtual keyboards used in e banking applications.

Unfortunately, virtual keyboards are not immune to attacks [2]. In user based attacks, an attacker can gather passwords by observ ing the target screen from a distance (shoulder surfing attacks [16]). With respect to software based attacks, the early versions of malware disclosed passwords by capturing the coordinates of the mouse on each click [3]. These attacks can be prevented by:

* Corresponding author at: Department of Computer Science and Engineering, University Carlos III of Madrid. Av. de la Universidad, 30, 28911, Leganés, Madrid, Spain.

E-mail address: pperis@inf.uc3m.es (P. Peris-Lopez).

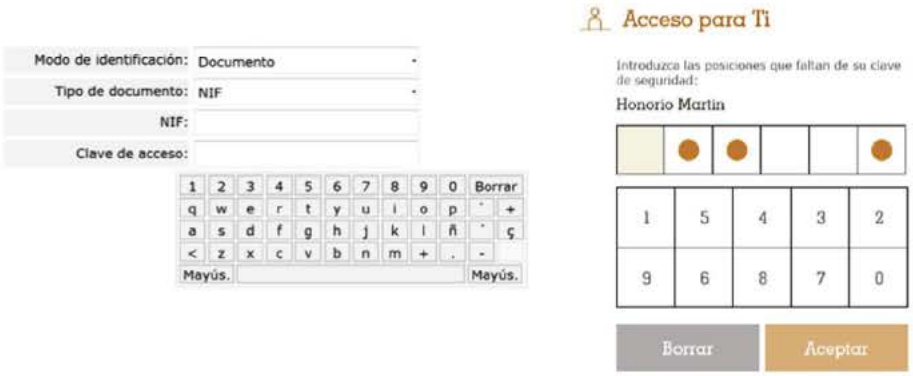

Fig. 1. Virtual keyboards in e-banking.

1) mutating the position of the virtual keyboard on every click [6]; 2) using obfuscation techniques [4]; and 3) hiding the position of the keys [4]. Nevertheless, Trojans (malicious software) have been designed to overpass these countermeasures. A video can record the user activity on the screen, or screenshots can be taken on each click [2,17].

The aforementioned malicious software must be installed and remain undetected to success in the attack. Nevertheless Hardware Trojans are more harmful since they are part of the device (hard ware) itself. In detail we propose a Hardware Trojan that targets a VGA display and is able to disclose the sensitive information clicked on a virtual keyboard. The rest of the paper is organized as follows. In Section 2 the main properties of Hardware Trojans are studied. In Section 3 we present a proof of concept of a simple but effective Hardware Trojan against virtual keyboards. Finally conclusions are extracted in Section 4.

2. Hardware Trojans

A Hardware Trojan (HT) refers to a malicious modification of the hardware during design or fabrication [18,19]. As consequence of this, the functional behaviour of the Integrated Circuit (IC) is altered, which puts at risk its security. In the literature we can find several classifications of HTs likes the ones presented in [20 22]. The high level taxonomy introduced in [20] is enough to under stand the principles of HTs. In this classification, HTs are cataloged taking into account two main features (i.e., activation mechanism and payload) see Fig. 2 for details. We briefly describe each of them:

- *Activation mechanism.* Depending on the trigger condition, HTs can be categorized into analog or digital triggered. Among the first ones stand out those trojans that are activated by operating conditions such as temperature, delays or electromagnetic waves. On the other hand, digital triggered trojans are activated using a rare boolean logic function. Regarding the nature of this function, HTs can be further classified into combinational or sequential.
- *Payload.* HTs can be cataloged regarding their disrupted opera tion into three main categories: digital, analog and others. Among the most common and extended payloads we can find HTs that attempt somehow to leak secret information. The aforesaid secret information is transmitted using the communi cation channels already integrated into the original design or by using a side channel such as the power consumption or thermal radiation. On the other hand, another popular HT payload is the Denial of Service (DoS) attack, where a HT can render a service unavailable at a particular time.

Other parameters used to define and compare different Trojans include the physical characteristics. The hardware used by the HT is directly linked with its physical characteristics. Avoiding detec tion is crucial for the success of the Trojan. Two physical features (size and power) are commonly considered with the aim of bypass ing regular security controls. The Trojan size must be minimized as much as possible in order to avoid detection by visual inspection. This task is commonly done re using the existing parts in the orig inal IC and exploiting the high density of the circuits to hide the new components. With respect to the power consumption, the amount of power consumed by the Trojan must be insignificant compared to the total amount of energy consumed by the rest of the circuit.

As shown below, we present a proof of concept of a triggered HT that disclose private information inserted by the user on a vir tual keyboard. The Trojan has been designed to circumvent the common protection mechanisms (i.e., size and power consumption analysis techniques).

3. A proof-of-concept

As mentioned in Section 1 several software based attacks against virtual keyboards have been proposed in the literature [2,17,23]. In this section we take a step forward in this direction and propose a HT against virtual keyboards.

3.1. Experimental framework

We briefly introduce two concepts: the VGA standard the pro posed HT targets the VGA and a practical case of a virtual key board used for banking operations.

3.1.1. VGA display

Video Graphics Array (VGA) is an IBM graphics standard that is implemented in almost all the PCs [24]. A VGA display is controlled by three analog signals, which specify colors intensity (i.e., RGB, Red Green Blue), and two synchronizing signals for determining the horizontal and vertical position. In order to draw an image, the display sweeps lines starting with the top left corner. Lines are drawn horizontally from left to right, one after another from top to bottom. To control how the display draws the image, it is necessary to generate a HSYNC horizontal synchronization pulse after drawing each line, and a VSYNC vertical synchronization pulse after drawing all the horizontal lines. To illustrate this, the waveforms of a complete screen are shown in Fig. 3.

In this standard we can know each pixel color and, by extension, redraw a screen by paying attention to the RGB signals and

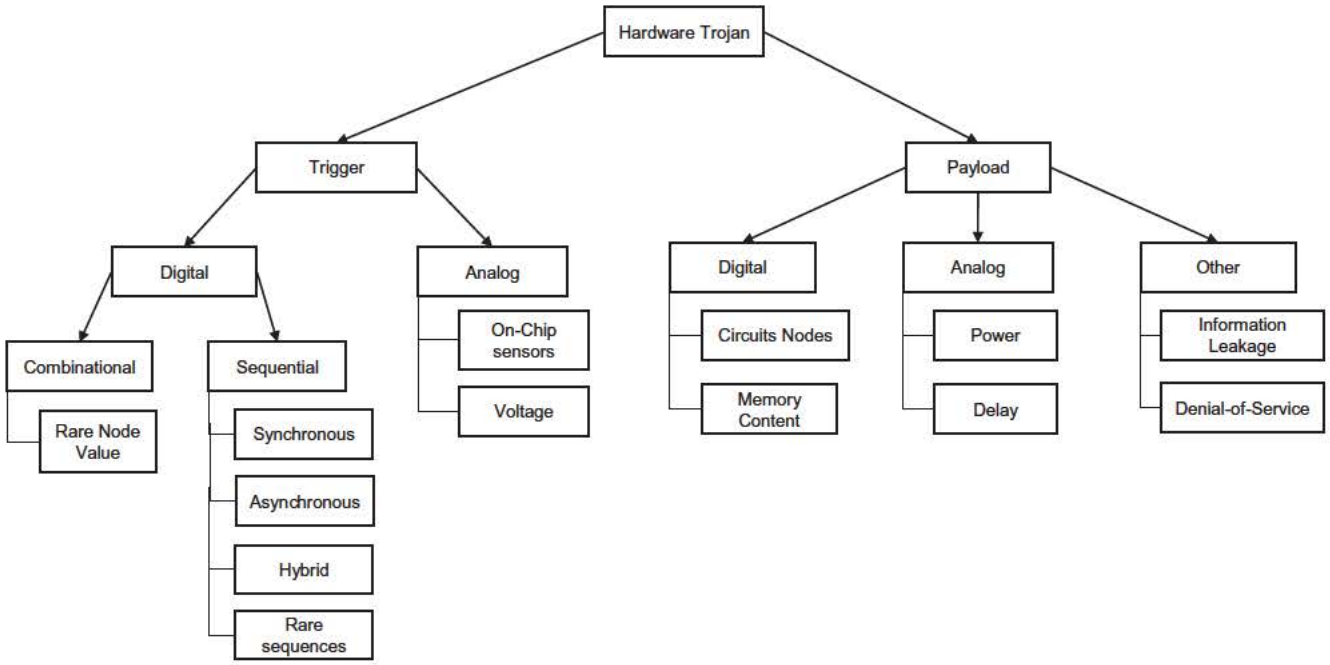

Fig. 2. Taxonomy for Hardware Trojans [20].

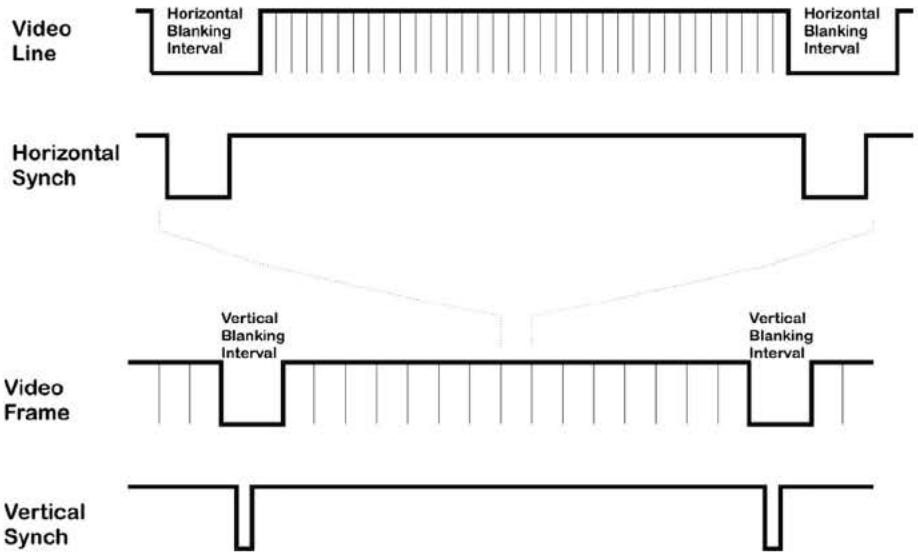

Fig. 3. Waveforms of a complete screen [25].

knowing the pixel clock and the screen resolution. For our purposes, we do not need to redraw the entire screen, only the virtual keyboard.

3.1.2. An example of a virtual keyboard

As an example of on line banking, which uses a virtual keyboard, we have chosen the ING Direct web page (Spanish version: https://ing.ingdirect.es/login/#pinpad). We have selected this platform due to two main reasons. First, the use of the virtual keyboard is mandatory in this web page there is not other way to introduce passwords. Second, two countermeasures are integrated into this platform to increase its security. The first one is the partial introduction of the password (only 3 positions of 6 are required each time). The second one entails a shuffling of the virtual keyboard in each authentication attempt. In Fig. 1, on the right side, the virtual keyboard under attack is shown.

3.2. Hardware Trojan attack

A typical industrial IC development process consists of many different phases where HTs can be injected. In order to describe our adversary model, we have selected the attack classification model presented in [26]. For the proposed HT, we have considered that an untrusted third party IP, that designs the VGA core, injects the HT (Model A). This attack model is very popular due to the proliferation of System on Chip (SoC) designs. Attack Model D, where an untrusted commercial off the shelf (COTS) model is contemplated, could also be considered the reader is urged to consult [26] for details. The designed HT targets the RGB signals of a VGA display. Fig. 4 depicts a typical VGA/LCD controller core [27]. This aforementioned core embeds a Wishbone interface that is in charge of the communication with other screen components (e.g., video memory). Among the different blocks, several registers,

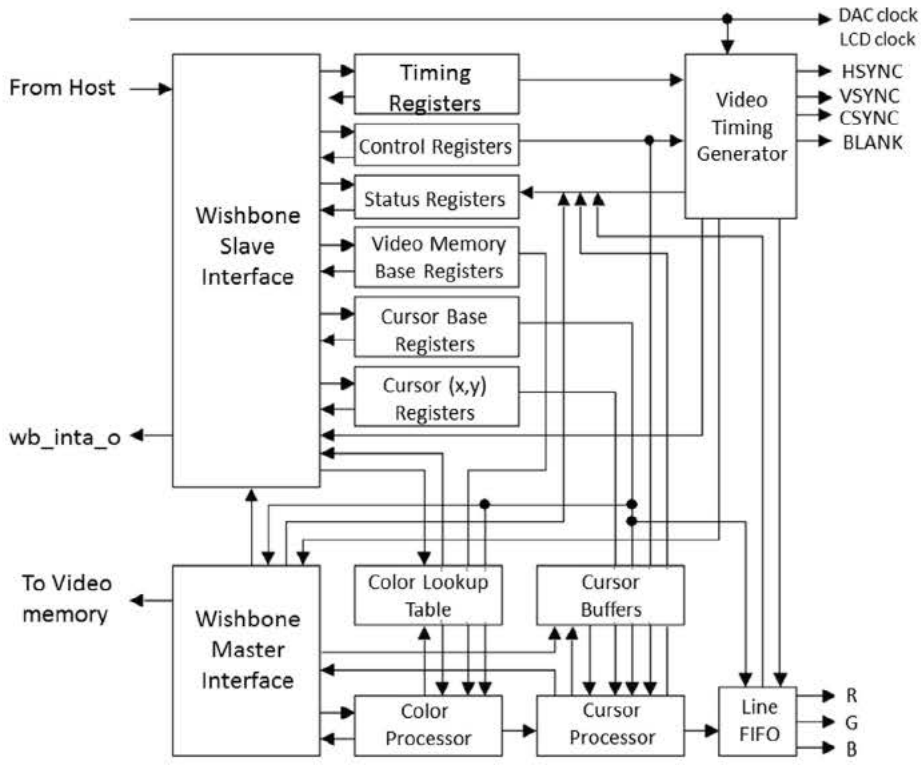

Fig. 4. Typical componets of a VGA/LCD controller core.

which are used to control different parameters related to the color and the timing of the display, stand out. The Color LookUp Table (CLUT) contains the RGB color information. The CLUT block is made up of a memory divided into two separate CLUTs (256×24 bits). Our proposed HT aims to capture and save the RGB signals in order to redraw the screen and finally retrieve the user's bank password. Further details of the HT are presented below following the classification described in Section 2.

- *Activation Mechanism:* The HT must be triggered when the bank webpage is displayed. To that end, the RGB values of the targeted webpage must be stored in order to compare these values with the current RGB values used to draw the screen. More specifically, it is sufficient if only the *Red* values are recorded. It is noteworthy that only a portion of the video frame is necessary to triger the HT. In order to overcome some reliability issues in connection with the activation mechanism, we have specified some parameters related to the screen: resolution, operating frequency, maximization or not of the browser window, and presence/absence of text in the video frame. Our HT proposal takes advantage of the reserved memory addresses of the VGA/LCD controller to save the desired sequence of RGB values. According to [27], these reserved locations cannot be accessed by the system: write accesses are ignored and read accesses return all zeros. For that, in the foundry, these memory positions would be initialized with the values of the targeted webpage and would undetected in the typical logic tests (testing of circuits). In order to avoid the synthesiser optimizations and guarantee the implementation of the reserved memory locations, we have employed a "*Keep*" attribute for these reserved locations. In addition, a comparator simply implemented with a bank of AND gates could be used to detect the webpage under attack. In summary, the HT is activated if the current RGB values are equal to those stored in the reserved memory. The main advantage of this trigger mechanism is that an attacker can interact with the HT to allow some flexibility in the configuration of the Trojan trigger (e.g., the initial values of the webpage could be changed via firmware update). In addition, as the reserved memory is already on board of the VGA/LCD core, the overhead of this activation mechanism is negligible.
- *Payload:* The final goal of the proposed HT is to retrieve the user password. To that end, we have modified the Wishbone interface. This modification lies in the addition of a new state that configures the Wishbone interface to store a screen shot into some specific addresses of the Video Memory. This state would only be reached when the HT is activated. For the data leakage, the adversary could exfiltrate the information through the On/Off LED that all monitors integrate the emitted light would be imperceptibly modulated for optical data transmission. Other possibility might be the data exfiltration through a firmware update. In this case, the adversary takes advantage of the update to access the content of the Video Memory and retrieve the recorded screen shots. The above mentioned approximations are very lightweight. Other possibilities include the use of conventional communication channels (e.g., NFC, Wifi, Bluetooth, etc.) or covert channels (e.g., modifying the packets transmission rate, using unused packet header fields, etc.) in order to reveal the user's password. The reader is urged to consult [28] or [29] for a detailed explanation of the commonly used cover channels.

3.3. Experimental results

3.3.1. Hardware implementation

As a proof of concept implementation we present a FPGA implementation (results can be extended to ASIC with some modifications). The re programmable nature of FPGAs makes easier the HT integration. More precisely, we have implemented a VGA/LCD

core ([27]) in a Spartan 3E. First we have synthesized and implemented the original core generating the NCD files of the clean model (without HT). After that we have initialized the reserved memory (i.e., 128 bits) with the values of the target webpage (in this case random values for testing purposes). In order to keep the same place and route, we have created a Hard macro containing AND gates of 4 inputs that will trigger the Trojan. We have inserted this hard macro in unused LUTs of the FPGA. Regarding the components of the core, firstly, we have modified the two CLUTs banks. Secondly, the alteration of the Wishbone Interface has been carried out by modifying the process accountable for the generation of addresses. In detail, we have added a new condition (Trojan trigger) in order to store few screen shots into specific addresses of the video memory. Since these addresses can not be accessed by the system (under the restrictions of the communication standard in the Wishbone interface), we can guarantee that the captured information will not be deleted by the normal functioning of the system. As a possible solution, we consider that the password will be exfiltrated through a firmware update no extra hardware is necessary for this purpose.

Exploiting the use of components already present in the display, it renders the proposed HT practically undetectable. In particular, the amount of used resources is negligible, making infeasible the detection by visual inspection. More specifically, the clean circuit uses 721 Slices and 645 registers while the infected circuit uses 752 Slices and 681 registers. This overhead represents an increment of 4.3% of Slices and 5.6% of registers. In addition, it is important to notice that *Red* values of the target webpage are stored into unused memory addresses that are implemented in LUTs.

In Fig. 5 both systems are depicted: the original, clean VGA core (left) and the VGA core with the HT inserted (right).

3.3.2. Operation

The proposed HT bypasses some countermeasures integrated in advanced virtual keyboards [4 6]. More precisely, this HT overpasses the key shuffling after each click on the virtual keyboard. Mutations in the order of the keys do not have consequences on our attack since we can reveal the whole keyboard and the clicked key independently of the order used. It also bypasses the anti screenshot technique that changes to some special symbol all the keys in a particular row of the keyboard when the mouse is moved to one key of that row. Note that when the target web site is detected, screenshots are taken at a rate enough to record all the modifications on the screen. Therefore, this hiding technique is also useless to impede our attack.

We have simulated the proposed HT using a logic analyzer (Gologic™ Analyzer) in order to show that it is possible to retrieve

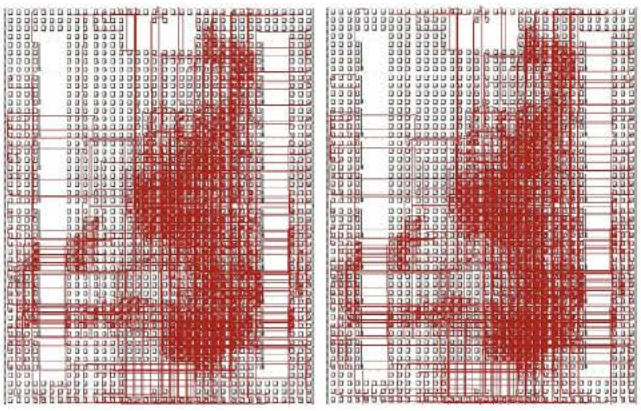

Fig. 5. Layout of the clean (left) and infected (right) VGA cores —implemented in a Spartan-3E.

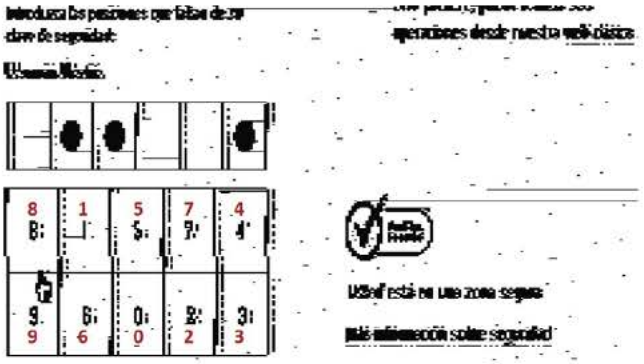

Fig. 6. Proof of concept: R signal captured by a logic analyzer.

the password with this technique. In this simulation we target the *Red* component of the RGB signal. We have selected a screen resolution of 640 × 480 (full screen) at 60 Hz and the targeted frame's portion does not contain letters. We have obtained and saved around 38 KB of data (one screenshot). In Fig. 6 we show the rebuild screen using the pixel information previously saved. As shown, numbers and the cursor can be clearly identified in particular, in this screeshot the number 9 key is clicked. Therefore, we can disclose the user's password using this technique. We just need to capture several screenshots to reveal the whole password. If the HT is not able to leakage all the information needed, it could be possible to use complex methods for image retrieval that are used for other technologies [30].

4. Conclusions

Virtual keyboards have been widely adopted in e banking applications in order to overcome security problems of physical keyboards. Nonetheless attackers might exploit the use of on screen keyboards in order to reveal sensitive information like passwords. In fact, several shoulder surfing and software based attacks have been proposed in the literature against them [2,3]. In response, some authors have proposed protection mechanisms to thwart these attacks [4 6]. In this paper we take a step forward and propose a Hardware Trojan that could be embedded in a VGA display. This HT can disclose private information each time the user clicked on the virtual keyboard. As an illustrative example, we have selected an e banking platform in which the use of the virtual keyboard is mandatory. We have scrutinized the main features of the HT in three distinct dimensions: activation mechanism, payload and physical characteristics. As a proof of concept we have implemented the HT on board of a VGA/LCD controller core and then simulated its functioning using a logic analyzer to show its devastating impact on the security of virtual keyboards. We emphasize here that the protection mechanisms commonly used in virtual keyboards are ineffective against this hardware attack since the HT can record all the changes on the display. Summarizing, HTs in VGA displays constitute a serious security risk for banks and users and, in general, for sensitive information inserted via a virtual keyboard.

Acknowledgments

This work was supported by the MINECO Spain grant: TIN2013 46469 R (SPINY: Security and Privacy in the Internet of You) and the CAM Comunidad Autónoma de Madrid grant S2013/ICE 3095 (CIBERDINE: Cybersecurity, Data, and Risks).

References

[1] Szopinski TS. Factors affecting the adoption of online banking in poland. J Bus Res 2016;69(11):4763–8.

[2] Nadkarni TS, Mohandas R, Pais AR. A novel technique for defeating virtual keyboards – exploiting insecure features of modern browsers. Advances in Computing and Communications, volume 191 of Communications in Computer and Information Science. Springer; 2011. p. 685–92.

[3] Sapra K, Husain B, Brooks R, Smith M. Circumventing keyloggers and screen dumps. Proc. of the 8th International Conference on Malicious and Unwanted Software: The Americas (MALWARE). p. 103–8.

[4] Nayak C, Parhi M, Ghosal S. Robust virtual keyboard for online banking. Int J Comput Appl 2014;107(21):36–8.

[5] Parekh A, Pawar A, Munot P, Mantri P. Secure authentication using anti-screenshot virtual keyboard. Int J Comput Sci Issues 2011;8(5):534–7.

[6] Rajarajan S, Maheswari K, Hemapriya R, Sriharilakshmi S. Shoulder surfing resistant virtual keyboard for internet banking. World Appl Sci J 2014;31 (7):1297–304.

[7] Caragata D, Mucarquer JA, Koscina M, El Assad S. Cryptanalysis of an improved fragile watermarking scheme. AEU – Int J Electron Commun 2016;70 (6):777–85.

[8] Boateng H, Adam DR, Okoe AF, Anning-Dorson T. Assessing the determinants of internet banking adoption intentions: A social cognitive theory perspective. Comput Human Behav 2016;65:468–78.

[9] Rachwald Rob. Is banking online safer than banking on the corner? Comput Fraud Secur 2008;2008(3):11–2.

[10] Gokhale AS, Waghmare VS. The shoulder surfing resistant graphical password authentication technique. Procedia Comput Sci 2016;79:490–8.

[11] Kiljan S, Vranken H, van Eekelen M. Evaluation of transaction authentication methods for online banking. Future Gener Comput Syst 2016.

[12] Seal A, Bhattacharjee D, Nasipuri M. Human face recognition using random forest based fusion of -trous wavelet transform coefficients from thermal and visible images. AEU – Int J Electron Commun 2016;70(8):1041–9.

[13] Aspinall D, Just M. Give me letters 2, 3 and 6!: partial password implementations and attacks. Financial Cryptography and Data Security, volume 7859 of Lecture Notes in Computer Science. Berlin Heidelberg: Springer; 2013. p. 126–43.

[14] Sagiroglu S, Canbek G. Keyloggers: increasing threats to computer security and privacy. IEEE Technol Soc Mag 2009;28:10–7.

[15] Damopoulos D, Kambourakis G, Gritzalis S. From keyloggers to touchloggers: take the rough with the smooth. Comput Secur 2013;32:102–14.

[16] Chakraborty N, Randhawa GS, Das K, Mondal S. Mobsecure: a shoulder surfing safe login approach implemented on mobile device. Procedia Comput Sci 2016;93:854–61. Proceedings of the 6th International Conference on Advances in Computing and Communications.

[17] Valeri L. Screen recording system for windows desktop. Russian-Korean International Symposium Science and Technology conf.. p. 107–9.

[18] Wang C, Li J, Yu M, Wang J. An intelligent classification method for trojan detection based on side-channel analysis. IEICE Electron Express 2013;10 (17):20130602.

[19] Li H, Liu Q, Zhang J. A survey of hardware trojan threat and defense. Integration, the VLSI Journal 2016;55:426–37.

[20] Bhunia S, Hsiao MS, Banga M, Narasimhan S. Hardware trojan attacks: threat analysis and countermeasures. Proc IEEE Aug 2014;102(8):1229–47.

[21] H. Salmani, M. Tehranipoor. https://www.trust-hub.org/benchmarks.php. In Trust HUB.org, Consulted on Sept. 2016

[22] Moein S, Gulliver TA, Gebali F, Alkandari A. A new characterization of hardware trojans. IEEE Access 2016;4:2721–31.

[23] Raza M, Iqbal M, Sharif M. A survey of password attacks and comparative analysis on methods for secure authentication. World Appl Sci J 2012;19 (4):439–44.

[24] Wilson Peter R. A simple VGA interface. In: Wilson PR, editor. Design Recipes for FPGAs. Newnes; 2007. p. 161–8.

[25] Vanden Bout D. Vga signal generation with the xs board. http://www.xess.com/static/media/appnotes/vga.pdf. Accessed: 2016-12-10

[26] Xiao K, Forte D, Jin Y, Karri R, Bhunia S, Tehranipoor M. Hardware trojans: Lessons learned after one decade of research. ACM Trans. Des. Autom. Electron. Syst. 2016;22(1):6:1–6:23.

[27] Richard Herveille. Vga/lcd core v 2.0 specifications. http://opencores.org/websvn,filedetails?repname=vgalcd&path=%2Fvgalcd%2Ftrunk%2Fdoc%2Fvgacore.pdf&rev=27. Accessed: 2016-12-10

[28] Heda Y, Shah R. Covert channel design and detection techniques: a survey. IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT). p. 1–6.

[29] Bachy Y, Basse F, Nicomette V, Alata E, Kaaniche M, Courrege JC, Lukjanenko P. Smart-tv security analysis: Practical experiments. 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks 2015:497–504.

[30] Charles Yesubai Rubavathi, Ramraj Ravi. Regular paper. AEUE – Int J Electron Commun 2016;70(3):225–33.